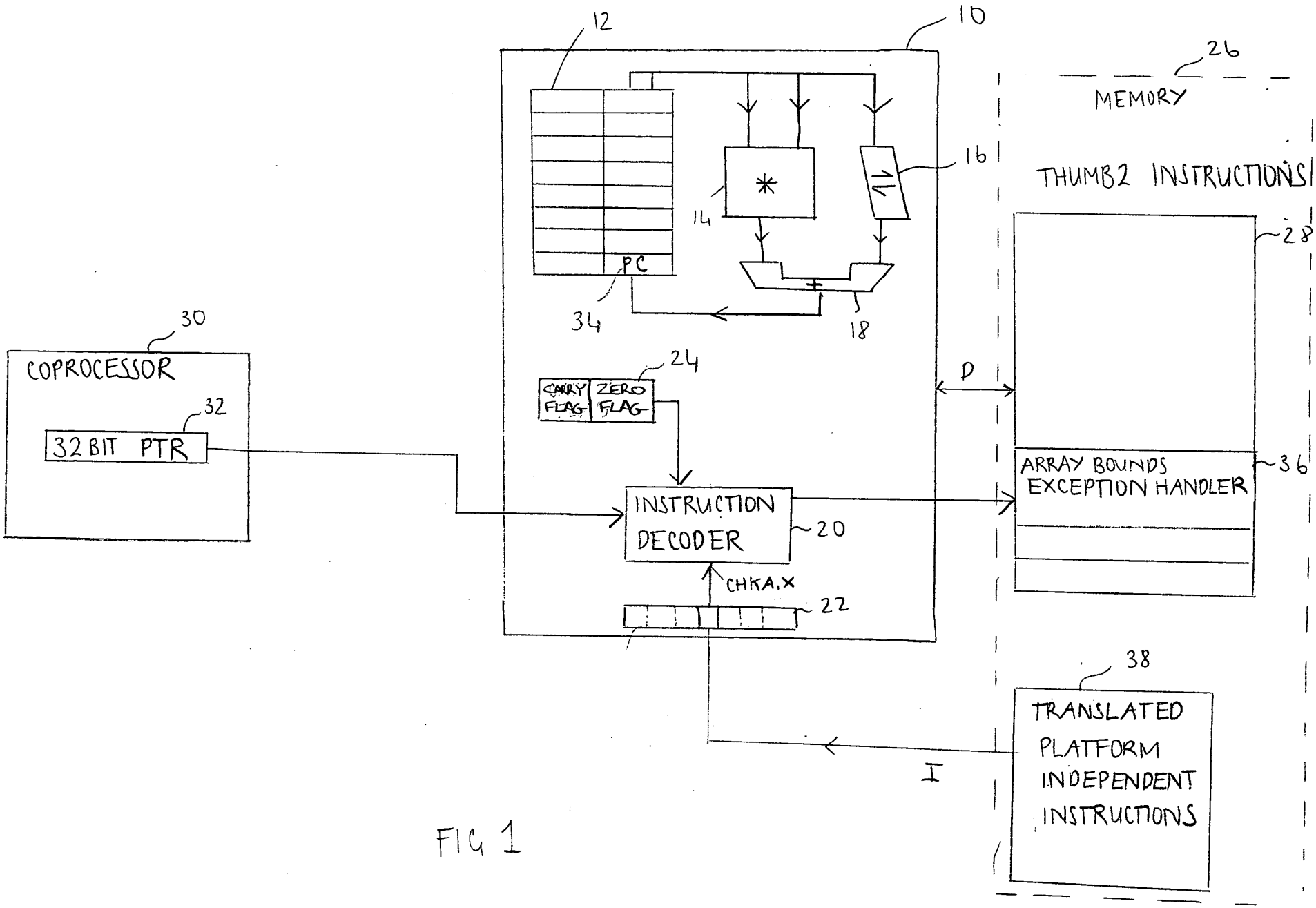
12
10
26
MEMORY
THUMB2 INSTRUCTIONS
16
14
28
PC
34
18
30
COPROCESSOR
24
CARRY FLAG
ZERO FLAG
D
32
32 BIT PTR
ARRAY BOUNDS EXCEPTION HANDLER
36
INSTRUCTION DECODER
20
CHKA.X
22
38
TRANSLATED PLATFORM INDEPENDENT INSTRUCTIONS
I

FIG 1

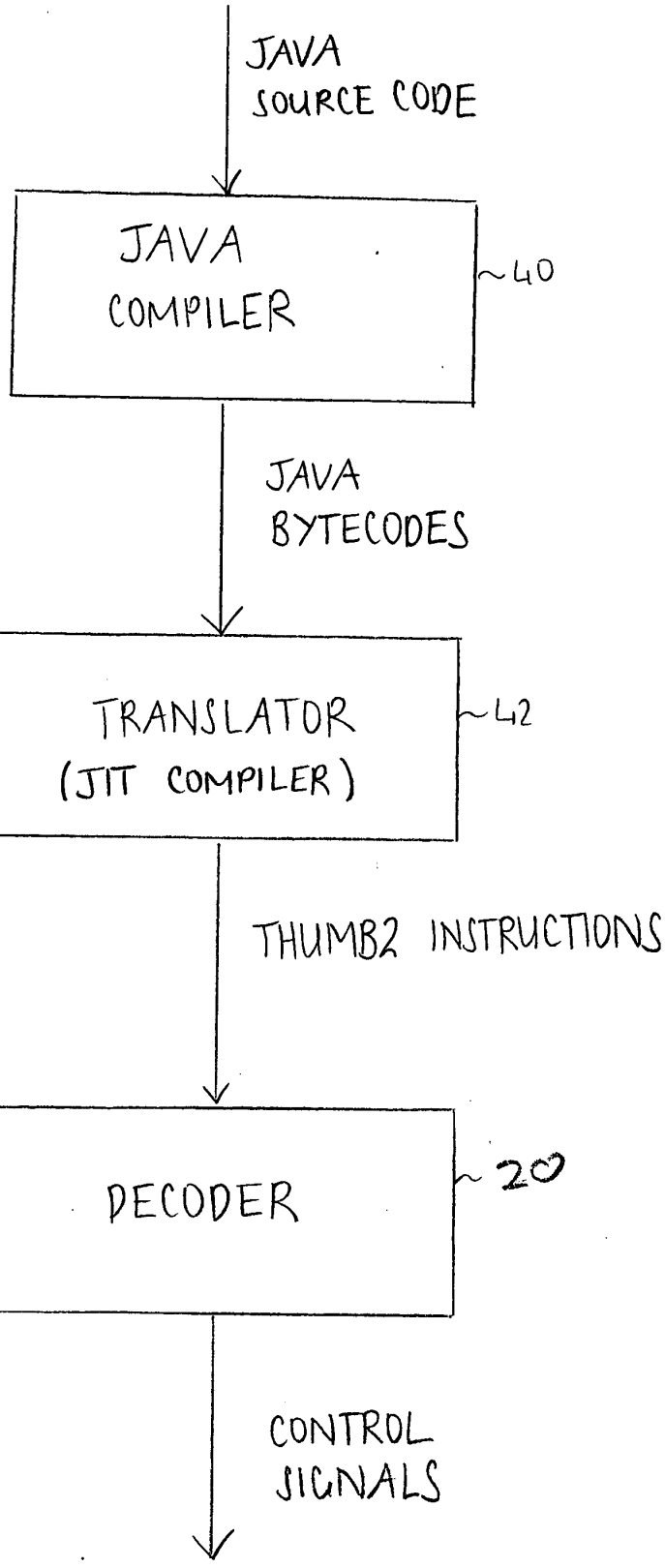
JAVA
SOURCE CODE
JAVA
COMPILER
40
JAVA
BYTECODES
TRANSLATOR
(JIT COMPILER)
42
THUMB2 INSTRUCTIONS
DECODER
20
CONTROL
SIGNALS

FIG 2

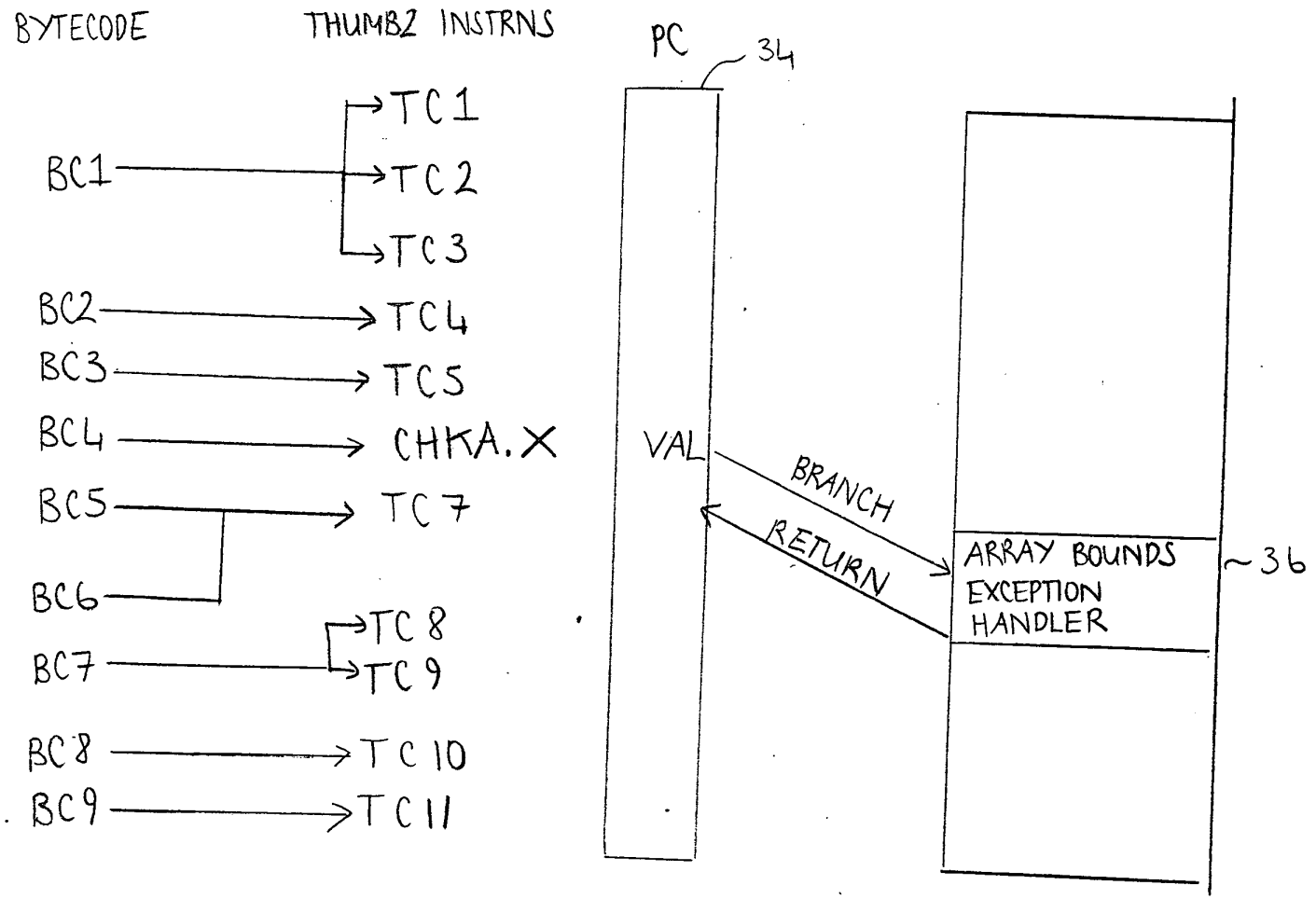
BYTECODE
THUMB2 INSTRNS
BC1
TC1
TC2
TC3
BC2
TC4
BC3
TC5
BC4
CHKA.X
BC5
TC7
BC6
BC7
TC8
TC9
BC8
TC10
BC9
TC11
PC
34
VAL
BRANCH
RETURN
ARRAY BOUNDS
EXCEPTION
HANDLER
36

FIG. 3

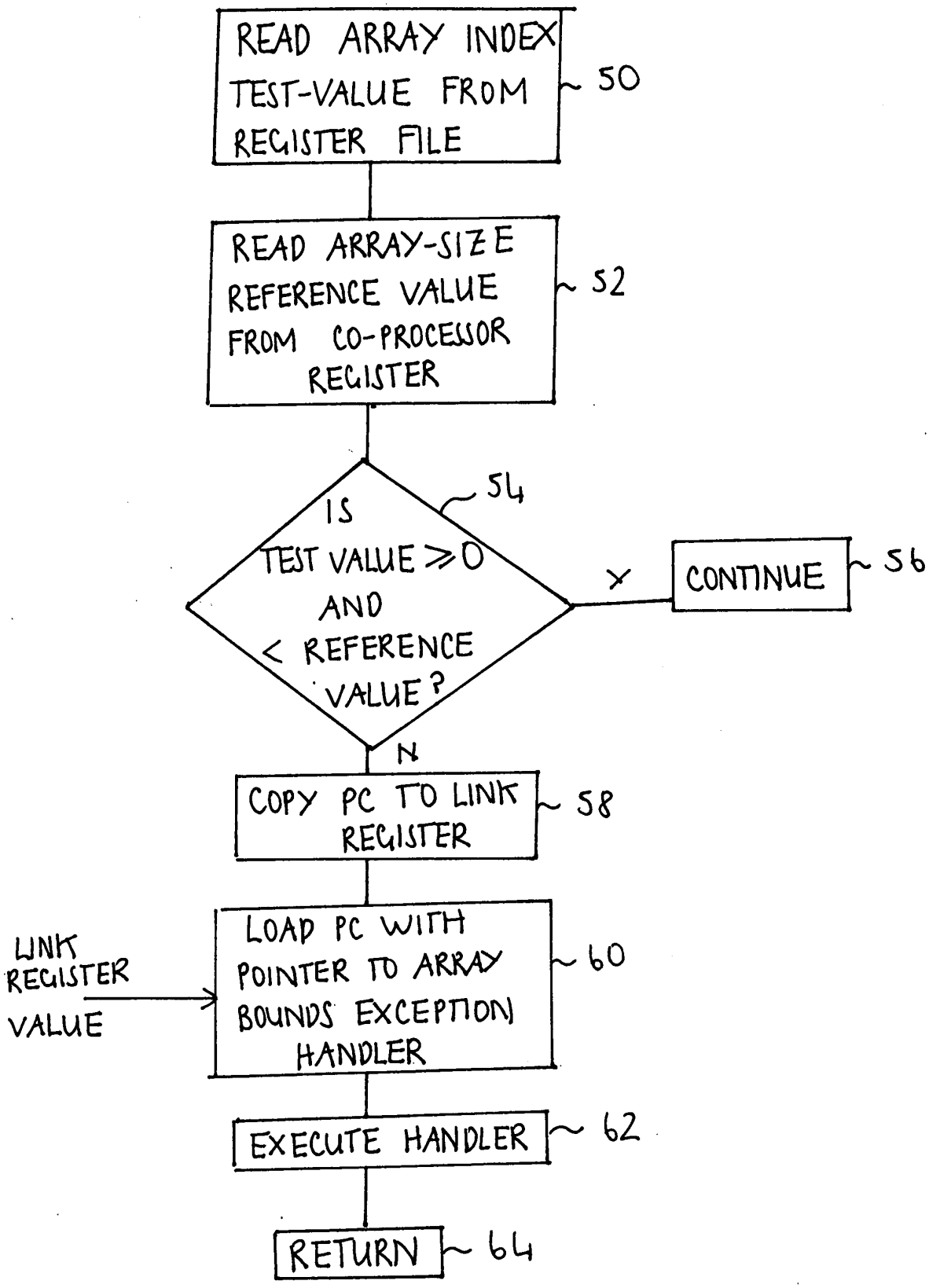
READ ARRAY INDEX TEST-VALUE FROM REGISTER FILE
50
READ ARRAY-SIZE REFERENCE VALUE FROM CO-PROCESSOR REGISTER
52
IS TEST VALUE ≥ 0 AND < REFERENCE VALUE?
54
Y
CONTINUE
56
N
COPY PC TO LINK REGISTER
58
LINK REGISTER VALUE
LOAD PC WITH POINTER TO ARRAY BOUNDS EXCEPTION HANDLER
60
EXECUTE HANDLER
62
RETURN
64

FIG 4

Instruction	CHKA.X Rn,Rm (16-bit)													
Encoding	15	14	13	12	11	10	9	8	7	6	5	3	2	0
	< op code >								H1	H2	Rm		Rn	
Thumb-2 Equivalent	CMP Rn,Rm MOVLS lr,pc ADDLS pc,HandlerBase,#-8													
Definition	IF (unsigned)Rm >= (unsigned)Rn lr = pc pc = HandlerBase,#-8 ; IndexException													
Encoding space	2^8								8 bits					
Note	This is based upon the CMP(3) 16-bit Thumb-2 instruction that can use high registers													
Note	H1 contains the most significant bit for Rn, H2 the most significant bit for Rm													
Note	The LS case should almost never occur, so can be treated as exceptional behaviour													
Note	This instruction dos not set condition flags													
Note	This comparison is UNSIGNED													
Note	Return stack prediction will not be required when the MOV lr,pc step is executed.													

FIG 5